A Survey of Distributed Computer Vision Algorithms

Richard J. Radke

Abstract Recent years have seen great advances in computer vision research dealing with large numbers of cameras. However, many multi-camera computer vision algorithms assume that the information from all cameras is losslessly communicated to a central processor that solves the problem. This assumption is unrealistic for emerging wireless camera networks, which may contain processors with limited capability, antennas with limited power, and batteries with limited life. In this chapter, we will overview algorithms that solve computer vision problems in a distributed manner, making them well-suited for implementation on a visual sensor network.

1 Introduction

Over the past twenty years, the computer vision community has made great strides in the automatic solution to such problems as camera localization and visual tracking. Many algorithms have been made tractable by the rapid increases in computational speed and memory size now available to a single computer. However, the world of visual sensor networks poses several challenges to the direct application of traditional computer vision algorithms. First, visual sensor networks are assumed to contain tens to hundreds of cameras- many more than are considered in many vision applications. Second, these cameras are likely to be spread over a wide geographical area- much larger than the typical computer lab. Third, the cameras are likely to have modest local processors with no ability to communicate beyond a short range. Fourth, and perhaps most importantly, no powerful central processor typically exists

Richard J. Radke
Department of Electrical, Computer, and Systems Engineering
Rensselaer Polytechnic Institute
Troy, NY, USA
e-mail: rjradke@ecse.rpi.edu

to collect the images from all cameras and solve computer vision problems relating them all.

Even if a central processor is available, the sheer number of cameras now feasible in a visual sensor network may preclude the use of a centralized algorithm. For example, more than half a million CCTV cameras observe the streets of London, with thousands in the airports and subways alone. Near real-time processing of this amount of data at a central processor (either with automatic vision algorithms or human observation) is simply infeasible, arguing for so-called “smart cameras” that can perform image processing and computer vision tasks locally before transmission to the broader network.

These trends all point to the need for *distributed* algorithms for computer vision problems in visual sensor networks. In addition to being well-suited to ad-hoc wireless networks, distributed algorithms have no single point of failure, generally make fairer use of underyling communication links and computational resources, and are more robust and scalable than centralized algorithms.

In this chapter, we survey the recent literature on distributed computer vision algorithms designed for implementation in a visual sensor network. We note that many algorithms described as “distributed” either require data to be transmitted to a central processor after some on-board processing at the sensor, or require each node to collect data from all other nodes in the network, effectively making every node perform the same central computation. While such research is valuable, in this chapter we emphasize techniques that are truly distributed: that is, nodes in the network make local decisions based only on information from their immediate neighbors. The design goal for such algorithms should be that the distributed result approaches what could be obtained had all the information been available at a central processor. We also do not consider algorithms that are “distributed” in the sense of being efficiently implemented in parallel on a central multi-core processor.

The algorithms we survey fall into a few main categories. Section 2 reviews general work on non-computer-vision related distributed algorithms, many of which have recently found new applications in wireless sensor networks. Section 3 discusses the problem of topology estimation in a visual sensor network- that is, the determination of which nodes share overlapping fields of view, or have corresponding entry/exit points. Section 4 discusses the distributed calibration of a camera network- that is, the estimation of the three-dimensional location and orientation of each camera with respect to a common coordinate system. Section 5 discusses a common application of visual sensor networks: the distributed tracking of single or multiple moving objects, which may also include identification or classification of the objects. Section 6 concludes the chapter with a discussion of future trends and opportunities.

We note that our interest here is at the application layer of a visual sensor network- that is, the computer vision algorithms running at each node as opposed to the physical layer of the sensor nodes. The algorithms described here are generally applicable to the many “smart camera” nodes under development in academia and industry that consist of a visual sensor, a set of embedded network and digital signal processors, and an antenna. An excellent taxonomy and historical overview of smart cameras was recently presented by Rinner et al. (Rinner, Winkler, Schriebl, Quaritsch, and Wolf, 2008).

Several chapters in the rest of this book go into more detail on specific instances of the algorithms generally surveyed here. We also note the increased academic and industrial interest in the area of distributed computer vision. Much of the most innovative work has appeared in the 2006 Distributed Smart Camera Workshop, and the 2007 and 2008 International Conference on Distributed Smart Cameras. Recent special issues of the Proceedings of the IEEE, the IEEE Journal on Selected Topics in Signal Processing, and the EURASIP Journal on Advances in Signal Processing dedicated to multi-camera networks further address the topics.

2 Distributed Algorithms

The basic theory of distributed algorithms is well-described in the textbook by Lynch (Lynch, 1996). Distributed algorithms are categorized by whether they are synchronous, asynchronous, or partially synchronous, and whether communication is established using shared memory, broadcast messages, or point-to-point communication. Typical problems include electing a leader, conducting a breadth-first search, finding shortest paths, constructing a minimum spanning tree, and reaching consensus. In general, provably convergent algorithms are more difficult to derive in asynchronous networks than in synchronous ones. However, most of the work surveyed in this chapter implicitly assumes at least partially synchronous networks, which may not be realistic for real-world visual sensor networks.

The book by Bertsekas and Tsitsiklis (Bertsekas and Tsitsiklis, 1997) discusses somewhat higher-level parallel and distributed algorithms in the context of numerical analysis and optimization. They consider linear and nonlinear asynchronous iterative methods for algebraic and differential equations, dynamic programming, unconstrained and constrained optimization, and variational inequalities. The book by Varshney (Varshney, 1997) treats classical detection theory and hypothesis testing in a distributed framework, including Bayesian, Neyman-Pearson, minimax, sequential, and locally optimum detection.

We note that many distributed estimation problems arising from visual sensor networks can be viewed as special types of consensus problems. That is, all nodes in the network should come to an agreement about global parameters describing the network, or objects moving in it, without communicating all sensor data to a centralized location. For example, all the cameras in the network might need to agree on the estimated position of a central landmark.

The past several years have seen an explosion of interest in distributed algorithms specifically designed for wireless ad-hoc sensor networks, often taking into account networking, communication, and power considerations.

A substantial body of work exists on the distributed localization of nodes in wireless sensor networks based on considerations such as the time-of-flight between nodes. For example, Iyengar and Sikdar (Iyengar and Sikdar, 2003) proposed a distributed, scalable algorithm for sensor node localization in the plane that used time-of-arrival measurements instead of GPS. Marinakis and Dudek (Marinakis and Dudek, 2006, 2008) presented several innovative probabilistic algorithms for sensor network localization based on noisy inter-sensor range data. Langendoen and Reijers (Langendoen and Reijers, 2003) overviewed and compared several algorithms for the localization problem. We can think of the algorithms discussed in Sections 3 and 4 as extensions of the localization problem in 3D that use visual information as the basis for estimation.

Similarly, these algorithms, as well as the tracking algorithms considered in Section 5, can be considered as instances of distributed parameter estimation or optimization problems, another topic of interest in the sensor networking community. For example, Boyd et al. (Boyd, Ghosh, Prabhakar, and Shah, 2005) overviewed gossip algorithms for information exchange in sensor networks. They showed how such algorithms could be applied to the distributed estimation of an average value in a wireless sensor network.

Nowak (Nowak, 2003) showed how the parameters of a mixture of Gaussians for which each node of a sensor network had different mixing coefficients could be estimated using a distributed version of the well-known expectation-maximization (EM) algorithm. This message-passing algorithm involves the transmission of sufficient statistics between neighboring nodes in a specific order, and was experimentally shown to converge to the same results as centralized EM. Kowalczyk and Vlassis (Kowalczyk and Vlassis, 2004) proposed a related gossip-based distributed algorithm called Newscast EM for estimating the parameters of a Gaussian mixture. Random pairs of nodes repeatedly exchange their parameter estimates and combine them by weighted averaging.

Rabbat and Nowak (Rabbat and Nowak, 2004) investigated a general class of distributed optimization algorithms for sensor networks. They showed that many natural optimizations could be solved incrementally, by each node updating the estimated parameters it receives to reduce its local cost and passing the updated parameters on. Using the theory of incremental subgradient optimization, they showed that the distributed algorithm converges to within a small amount of the globally optimal value after only a few cycles through the network. They applied the approach to robust least-squares estimation, energy-based source localization, and decision boundary problems.

Finally, we note research on activating or orienting directional sensors to maximize spatial coverage. Ai and Abouzeid (Ai and Abouzeid, 2006) described a distributed algorithm for maximizing spatial coverage in an ad-hoc network of directional sensors. They proposed a distributed greedy algorithm that compared favorably to a central greedy algorithm, as well as a dynamic sensor scheduling algorithm that favorably distributed its energy requirements. Hoffmann and Hähner (Hoffmann and Hähner, 2007) proposed a simple local algorithm to maximize visual coverage in a network of smart cameras. They simulated their results on networks of up to 80 cameras, and analyzed the underlying networking issues using the NS2 network simulator. Bash and Desnoyers (Bash and Desnoyers, 2007) proposed an algorithm for the distributed exact computation of Voronoi cells in sensor networks, which could be useful for addressing coverage and tracking problems.

3 Topology Estimation

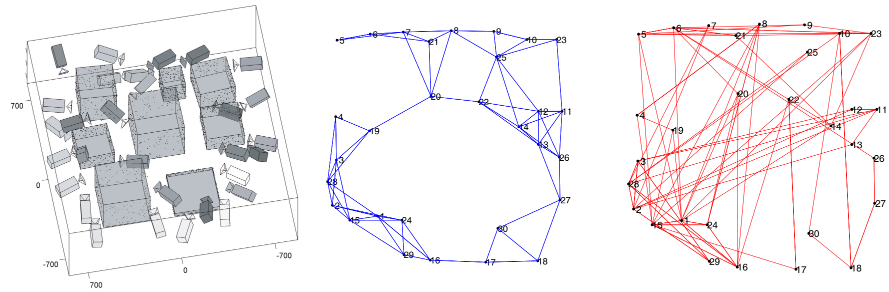
We consider two main types of topology estimation problems in visual sensor networks. The first is what we refer to as the *overlapping* problem, where it is assumed that the cameras observe parts of the same environment from different perspectives. The relationships between the cameras can be modeled as an undirected graph in which an edge appears between two cameras if they observe some of the same scene points from different perspectives. In our research, we called this the *vision graph* (Cheng, Devarajan, and Radke, 2007). We might naturally expect cameras' fields of view to overlap in wireless ad-hoc camera networks, in which a large number of cameras are densely distributed around a large, open environment.

The second topology estimation problem is what we refer to as the *non-overlapping* problem, where it is assumed that no two cameras observe the same part of the environment. Instead, relationships between the cameras are induced by the likelihood that an object in one camera appears in another after some amount of time. The relationships between the cameras can be again be modeled as a graph, but in this case a directed graph with expected transition probabilities is a more natural model (for example, it may be much more likely for objects to travel from Camera A to B than vice versa). This situation naturally arises in indoor surveillance scenarios, such as cameras mounted in corridors. For this reason, most research on non-overlapping networks assumes a planar environment.

In either case, we should keep in mind that the topology graph estimation problem in a wireless visual sensor network is accomplished by sending messages along underlying one-hop communication links. These links can be abstracted as a *communication graph* (Haas et al, 2002), which is mostly determined by the locations of the nodes, the powers of their antennas, and the topography of the environment. The camera topology estimation problem can be viewed as an *overlay graph* on this network.

We note that the presence of an edge in the communication graph does not imply the presence of the same edge in the vision graph, since nearby cameras may be pointed in different directions, and conversely, physically distant cameras can image part of the same scene. Figure 1 illustrates the communication and vision graphs for a simulated example in which 30 cameras are scattered around several buildings. We can see that the communication and vision graphs are not highly correlated.

As Figure 1 illustrates, there is generally no way to rule out the possibility of overlap between any two visual sensors without comparing their images (or extracted features from them). Consequently, correctly estimating the camera network topology inherently involves collecting the information from all sensors in one place, either at a central node, or at all nodes in parallel. We review several such methods here, emphasizing those designed with the constraints of a visual sen-



(a) (b) (c)

Fig. 1 (a) A simulated camera network (the focal lengths of the cameras have been exaggerated). (b) The corresponding communication graph, assuming each camera has the same fixed antenna range. (c) The corresponding vision graph.

sor network in mind. Even though such algorithms may be better characterized as decentralized than distributed, they are an important first step for the higher-level vision problems we discuss subsequently.

We review approaches to the topology estimation problem for both cases below. Virtually all research in the area seems to cast itself in one domain or the other- that is, either the cameras are viewed as entirely non-overlapping, or non-overlapping cameras are not considered in the topology. The paper by Wang et al. (Wang, Sung, and Ng, 2002) is an exception that allows for no, little, or large overlaps between fields of view, but the focus is on activity analysis of multi-camera trajectories rather than topology estimation.

3.1 Non-overlapping Topology Estimation

The non-overlapping topology estimation problem is well-studied. Most researchers have approached the problem using the spatio-temporal paths of tracked objects that enter and exit the field of view of each camera. Marinakis and Dudek (Marinakis and Dudek, 2005) described how to estimate a Markov model characterizing transitions between cameras, using only non-discriminative detected motions as input. They reconstructed object trajectories that best explain the cameras' observations, and used these to estimate the transition graph. The number of objects moving through the camera network was assumed known, and their probabilistic behavior was assumed to be homogeneous. Makris et al. (Makris, Ellis, and Black, 2004) proposed a similar approach that temporally correlates incoming and outgoing targets in detected entry and exit zones for each camera. Tieu et al. (Tieu, Dalley, and Grimson, 2005) used mutual information as a measure of statistical dependence to estimate object correspondences for topology estimation. Special cases of objects used for non-overlapping topology estimation include identifiable people (Zou, Bhanu, Song, and Roy-Chowdhury, 2007) and vehicles (Niu and Grimson, 2006).

Mandel et al. (Mandel, Shimshoni, and Keren, 2007) described a simple algorithm for estimating camera network topology based on broadcast messages regarding time-stamped detected motions in different spatial regions. They applied a sequential probability ratio test (SPRT) to accept or reject the possibility that two cameras observe the same scene based on the correspondence (or lack thereof) of the aggregated detections. The use of the SPRT is a noteworthy idea for decision problems in sensor networks. Van den Hengel et al. (van den Hengel, Dick, Detmold, Cichowski, and Hill, 2007) took a slightly different approach they called exclusion, in which the vision graph was initially assumed to be fully connected, and edges were removed that are contradicted by observed evidence over time. They tested their algorithm on a network of 100 real cameras.

Javed et al. (Javed, Rasheed, Shafique, and Shah, 2003) incorporated the appearance of objects as well as their spatio-temporal motion into the topology estimation problem. While their emphasis was on tracking, they learned the topology of a set of cameras during a training phase that used Parzen windows to model the joint density of entry/exit times, locations, and velocities, and a Gaussian model for the Bhattacharyya distances between histograms of corresponding objects. They later extended their approach (Javed, Shafique, and Shah, 2005) to more accurately model the brightness transfer function relating pairs of cameras. It would be interesting to see how this approach scales to large networks of cameras.

Farrell and Davis (Farrell and Davis, 2008) proposed a different decentralized algorithm for topology discovery based on an information-theoretic appearance matching criterion on tracked objects moving through the network. More distinctive objects contribute proportionally more to the estimated transition model between cameras. This idea seems interesting, but it was not tested with real images.

Medeiros et al. (Medeiros, Park, and Kak, 2007) took a slightly different approach to estimating the relationships between cameras based on what they termed event-driven clustering. Instead of explicitly constructing a graph representing the camera network topology, they used estimated detections of the same object to construct clusters of the vision graph. The proposed protocol addresses cluster formation, merging, splitting, and interaction in a distributed camera network.

An alternate method for distributed feature matching (which implicitly defines a vision graph) was described by Avidan et al. (Avidan, Moses, and Moses, 2007), who used a probabilistic argument based on random graphs to analyze the propagation of wide-baseline stereo matching results obtained for a small number of image pairs to the remaining cameras.

Farrell et al. (Farrell, Doermann, and Davis, 2007) described a Bayesian framework for learning higher-order transition models in camera networks. That is, in addition to estimating the adjacency of cameras, the method can help answer questions such as “what fraction of objects passing through cameras X and Y will at some later time reach camera Z?”. They claimed that the higher-order models can improve tracking and inference performance. It would be interesting to see the implementation of this idea in a real network of cameras.

3.2 Overlapping Topology Estimation

When the images from the cameras overlap, the topology estimation problem is often combined with the calibration problem discussed in the next section. Here we mention several techniques that "stop" at the point of topology estimation.

Kulkarni et al. (Kulkarni, Shenoy, and Ganesan, 2007) proposed a method using the time-stamped appearances of a moving reference object to provide an approximate initialization for the cameras' regions of overlap, which is indirectly related to their topology. Similar approaches applied to the camera calibration problem are discussed in the next section. However, for large-scale visual sensor networks, we believe it is important to develop techniques in which neither the cameras nor the scene are altered from their natural state, with no assumptions about camera synchronization or interactions between the user and the environment.

Cheng et al. (Cheng, Devarajan, and Radke, 2007) described an algorithm for estimating the vision graph based on a fixed-length "feature digest" of automatically detected distinctive keypoints that each camera broadcasts to the rest of the network. These keypoints arise from the popular and successful Scale-Invariant Feature Transform (SIFT) detector/descriptor proposed by Lowe (Lowe, 2004). Each sending camera makes a tradeoff between the number of keypoints it can send and how much their descriptors are compressed, assuming a fixed bit budget for their message. Each receiver camera decompresses the feature digest to recover the approximate feature descriptors, which are matched with its own features to generate putative matches. If enough matches are found, a vision graph edge is established. They analyzed the tradeoffs between the size of the feature digest, the number of transmitted features, the level of compression, and the overall performance of the edge generation. This method is appealing since it requires no user interaction with the environment, using static scene features as the basis for edge generation.

4 Camera Network Calibration

Formally, *camera calibration* is the estimation of the parameters that describe how a perspective camera projects 3-D points in a fixed world coordinate system to 2-D points on an image plane. These calibration parameters can be divided into four or more *internal* parameters related to a camera's optics, such as its principal point, aspect ratio, lens distortion, and focal length, and six *external* parameters, namely the rotation angles and translation vector that relate the camera coordinate frame to the world coordinate frame. Since the fixed internal parameters of a given camera can usually be estimated individually prior to deployment (Collins and Tsin, 1999; Hartley, 1994; Stein, 1995), most research is mainly concerned with estimating the external parameters.

The classical problem of externally calibrating a pair of cameras is well-understood (Tsai, 1992); the parameter estimation usually requires a set of feature point correspondences in both images. When no points with known 3-D locations in the world coordinate frame are available, the cameras can be calibrated up to a similarity transformation (Hartley and Zisserman, 2000). That is, the cameras' positions and orientations can be accurately estimated relative to each other, but not in an absolute sense. Without metric information about the scene, an unknown scale parameter also remains; for example, the same set of images would be produced by cameras twice as far away from a scene that is twice as large.

Multi-camera calibration can be accomplished by minimizing a nonlinear cost function of the calibration parameters and a collection of unknown 3-D scene points projecting to matched image correspondences; this problem is also known as *Structure from Motion (SFM)*. The optimization process for estimating the parameters is called *bundle adjustment* (Triggs, McLauchlan, Hartley, and Fitzgibbon, 2000). Good results are achievable when the images and correspondences are all accessible to a powerful, central processor. It is beyond the scope of this chapter to survey the centralized SFM problem here, but the book by Hartley and Zisserman (Hartley and Zisserman, 2000) is an excellent reference. Also, Ma et al. (Ma, Soatto, Košecká, and Sastry, 2004) outlined a complete step-by-step recipe for the SFM problem, assuming the information from all cameras is collected in one place. We note that SFM is closely related to a problem in the robotics community called *Simultaneous Localization and Mapping (SLAM)* (Leonard and Durrant-Whyte, 1991; Thrun, Burgard, and Fox, 2005), in which mobile robots must estimate their locations from sensor data as they move through a scene.

4.1 Non-overlapping Camera Calibration

The algorithms discussed in this section generally assume a camera network with overlapping fields of view, as discussed in Section 3.2. In the non-overlapping camera case, estimating the actual 3D positions of the cameras based only on their visual information is generally a poorly-posed problem. That is, multiple cameras must actually observe many of the same 3D points as the basis for accurate estimation of localization parameters.

However, several researchers have proposed solutions that make assumptions about the dynamics of objects transiting the environment to get rough estimates of camera positions and orientations. In this area, the work by Rahimi et al. (Rahimi, Dunagan, and Darrell, 2004) is the most notable. They encoded prior knowledge about the state (location and velocity) of targets as a linear Gaussian Markov model, and then combined this prior with measurements from the camera network to produce maximum a posteriori estimates of global trajectories and hence camera calibration parameters. The approach was illustrated with networks of four to ten downward-facing cameras observing a single target moving on a ground plane. It would be interesting to see how this method generalizes to large numbers of cameras in a fully 3-D environment. However, it is not reasonable to expect that highly accurate calibration parameters could be obtained with this type of approach using naturally-moving objects in a real-world, well-separated camera network due to difficult-to-model dynamics and unmodeled sources and sinks.

Rekleitis et al. (Rekleitis and Dudek, 2005; Rekleitis, Meger, and Dudek, 2006) proposed the use of a mobile robot holding a calibration pattern (a checkboard or an array of unique fiducial markers) as the basis for non-overlapping camera calibration. The robot's pose on the ground plane, obtained from odometry, is used to provide each camera with the 3D positions of points on the calibration target.

4.2 Overlapping Camera Calibration

Several methods, while still centralized, have used visual sensor networks as the motivation for the design of calibration algorithms. These algorithms are usually based on different cameras' observations of the same object or objects in their environment. For this purpose, several researchers have suggested modulated light sources moved throughout a darkened room (Baker and Aloimonos, 2000; Chen, Davis, and Slusallek, 2000; Barton-Sweeney, Lymberopoulos, and Savvides, 2006) or placed on the cameras themselves (Taylor and Shirmohammadi, 2006). Maas (Maas, 1999) used a moving reference bar of known length, while Baker and Aloimonos (Baker and Aloimonos, 2003) used grids of parallel lines and square boxes. Liu et al. (Liu, Kulkarni, Shenoy, and Ganesan, 2006) proposed an algorithm in which each camera independently calibrates itself based on images of a placed reference device with known world coordinates. As mentioned previously, algorithms that require no user interaction with the environment are generally preferable for large-scale ad-hoc deployments.

Similar to the non-overlapping topology estimation problem, several researchers have proposed to obtain corresponding points for calibration by matching trajectories of objects that naturally move through the cameras' environment. Lee et al. (Lee, Romano, and Stein, 2000) described one such approach in which pedestrians and vehicles moving on a ground plane were used to calibrate two or three overlapping surveillance cameras. Other approaches along this line were proposed by Jaynes (Jaynes, 1999) and Meingast et al. (Meingast, Oh, and Sastry, 2007).

Several researchers have addressed distributed versions of this type of camera network calibration. For example, Lee and Aghajan (Lee and Aghajan, 2006) used the joint observations of a moving target to solve a set of nonlinear equations for its position (and hence, the camera positions) using a distributed version of the Gauss-Newton method. Another excellent algorithm of this type called SLAT (Simultaneous Location and Tracking) was proposed by Funiak et al. (Funiak, Guestrin, Paskin, and Sukthankar, 2006). The key innovations were a relative overparameterization of the cameras that allows Gaussian densities to be used, and a linearization procedure to address the uncertainty in camera angle for proper use of a Kalman filter. It would be very interesting to see how such an approach would scale to full calibration of large outdoor camera networks with many moving objects.

Many distributed algorithms for full 3D camera network calibration rely on using image correspondences to estimate the epipolar geometry between image pairs. When the intrinsic camera parameters are known, this information can be used to extract estimates of the rotation and translation between each camera pair. For example, Barton-Sweeney et al. (Barton-Sweeney, Lymberopoulos, and Savvides, 2006) used an Extended Kalman Filtering framework on the estimated epipolar geometry to estimate the rotation and translations between a camera pair. Mantzel et al. (Mantzel, Choi, and Baraniuk, 2004) proposed an algorithm called DALT (Distributed Alternating Localization and Triangulation) for camera network calibration. As the the title suggests, each camera calibrates itself by alternating between 1) triangulating 2D image projections into 3D space, assuming the camera matrices are known, and 2) estimating the camera matrices based on putative image-to-world correspondences. While their optimization approach is simpler than bundle adjustment, this work includes some interesting analysis of the computational and power requirements that would be incurred by the algorithm on a realistic platform. These algorithms used a user-inserted calibration object as the basis for establishing point correspondences.

Devarajan et al. (Devarajan and Radke, 2004; Devarajan, Radke, and Chung, 2006) proposed a calibration algorithm in which each camera only communicates with the cameras connected to it by an edge in the vision graph, exchanging image projections of common 3D points. After performing a local projective-to-metric factorization and bundle adjustment, each camera has an estimate of 1) its own location, orientation, and focal length, 2) the corresponding parameters for each of its neighbors in the vision graph, and 3) the 3D positions of the image feature points it has in common with its neighbors. They observed that the distributed calibration algorithm not only resulted in good calibration accuracy compared to centralized bundle adjustment, but also that the messages per communication link required to calibrate the network were more fairly distributed, implying a longer node lifetime.

Medeiros et al. (Medeiros, Iwaki, and Park, 2008) proposed a distributed cluster-based calibration algorithm for which they observed a similar result: the distributed algorithm required significantly fewer transmit and receive bits than for a centralized algorithm, while having comparable accuracy. However, their approach makes a restrictive assumption that distinctive target objects with fixed features at a known scale are used for calibration.

We note that Brand et al. (Brand, Antone, and Teller, 2004) presented a novel approach to large-scale camera network calibration using a graph embedding formalism. They first estimated node-to-node direction vectors in 3D using corresponding features and vanishing points (the signs and lengths of the vectors were unknown). They then showed how the solution to a simple eigenvalue problem resulted in the embedding of 3D camera locations that was maximally consistent with the measured directions. The algorithm is remarkably fast and accurate, albeit centralized. It would be very interesting to investigate whether the eigenvalue problems can be solved in a distributed manner.

4.3 Improving Calibration Consistency

Many researchers have proposed methods for dealing with inconsistent parameter estimates in wireless sensor networks. These are typically based on verifying whether estimates of the same parameter are sufficiently close (Sastry, Shankar, and Wagner, 2003; Capkun and Hubaux, 2005; Wei, Yu, and Guan, 2007) and/or consistent with prior knowledge (Du, Fang, and Peng, 2006). Most such approaches are well-suited for sensors that measure a scalar quantity such as temperature or pressure. In contrast, camera networks contain inconsistencies in a continuous, high-dimensional parameter space, and require principled statistical methods for resolving them.

Devarajan and Radke (Devarajan and Radke, 2007) proposed a calibration refinement method based on belief propagation (Murphy, Weiss, and Jordan, 1999), a method for probabilistic inference in networks that several researchers have recently applied to discrete or low-dimensional sensor networking and robotics problems (e.g., (Venkatesh, Alanyali, and Savas, 2006; Christopher and Avi, 2003; Thorpe and McEliece, 2002; Ihler, Fisher, Moses, and Willsky, 2004)). The belief propagation algorithm involved passing messages about common camera parameters along the edges of the vision graph that iteratively brought inconsistent estimates of camera parameters closer together. They discussed solutions to several challenges unique to the camera calibration problem. They showed that the inconsistency in camera localization was reduced by factors of 2 to 6 after applying their belief propagation algorithm, while still maintaining high accuracy in the parameter estimates.

Several researchers have recently applied more sophisticated and robust distributed inference algorithms than belief propagation to sensor fusion problems; the work of Paskin, Guestrin, and McFadden (Paskin and Guestrin, 2004; Paskin, Guestrin, and McFadden, 2005) and Dellaert et al. (Dellaert, Kipp, and Krauthausen, 2005) is notable. Future distributed camera calibration techniques could benefit from these new algorithms.

Some researchers have suggested improvements in calibration by augmenting the input from visual sensors with non-visual sensor measurements. For example, Meingast et al. (Meingast, Kushwaha, Oh, Koutsoukos, Ledeczi, and Sastry, 2008) fused image data with radio interferometry to estimate visual sensors' position and orientation. However, this method relied on centralized processing.

5 Tracking and Classification

A visual sensor network is usually calibrated for an application-level purpose. Commonly, the camera network is tasked to track and/or classify objects as they move through the environment. A massive literature on single-camera or centralized multi-camera tracking exists; here we focus on distributed algorithms for the tracking problem designed with sensor networking considerations in mind.

Rao and Durrant-Whyte (Rao and Durrant-Whyte, 1993) presented early work that recognized the importance of a decentralized algorithm for target tracking. They proposed a Bayesian approach, but required a fully connected network topology that is unrealistic for modern sensor networks. Hoffmann et al. (Hoffmann and Hähner, 2007) described a distributed algorithm for multi-camera tracking, modeled as a finite state machine. The proposed algorithm was validated in a simulated network in which it was assumed that object detection results were entirely accurate.

Dockstader and Tekalp (Dockstader and Tekalp, 2001) presented an algorithm for distributed real-time multi-object tracking, based on a Bayesian belief network formalism for fusing observations with various confidence estimates. The basic strategy is for each camera to track each object independently in 2D, apply a distributed fusion/triangulation algorithm to obtain a 3D location, and then independently apply a 3D Kalman filter at each camera. Results were demonstrated for a three-camera network. Qu et al. (Qu, Schonfeld, and Mohamed, 2007) proposed a Bayesian algorithm for distributed multi-target tracking using multiple collaborative cameras. Their approach was based on a dynamic graphical model that updates estimates of objects' 2D positions instead of tracking in 3D. The algorithm is well-suited for challenging environments such as crowds of similar pedestrians. Heath and Guibas (Heath and Guibas, 2008) described an algorithm for multi-person tracking in a network of stereo (i.e., two closely-separated) cameras. The use of stereo cameras enables each node to estimate the trajectories of 3D points on each moving object. Each node communicates these feature positions to its neighbors and performs independent 3D particle filtering based on its received messages. They demonstrated results on small networks of real cameras as well as a larger simulated camera network observing a crowded scene. They analyzed the tradeoffs relating the communication budget for each camera, the number of tracked people in the environment, and the tracking precision.

Heath and Guibas (Heath and Guibas, 2007) also described an algorithm for tracking people and acquiring canonical (i.e., frontal-facing) face images in a camera network. They proposed a tracking algorithm based on 2D visual hulls synthesized at clusterhead nodes, and used the direction of pedestrian motion as a simple estimate for head orientation. The idea seems promising and a more fully distributed system would be interesting.

Mensink et al. (Mensink, Zajdel, and Krose, 2007) described a multi-camera tracking algorithm in which the observation at each camera was modeled as a mixture of Gaussians, with each Gaussian representing the appearance features of a single person. The authors extended the gossip-based Newscast EM algorithm (Kowalczyk and Vlassis, 2004) discussed in Section 2 to incorporate multiple observations into the iterative, distributed computation of means, covariances, and mixing weights in the Gaussian mixture model. The preliminary results showed that the distributed estimation algorithm converged as quickly and accurately as a centralized EM algorithm. This is an excellent example of a distributed computer vision algorithm.

Leistner et al. (Leistner, Roth, Grabner, Bischof, Starzacher, and Rinner, 2008) described an interesting technique for automatic person detection that applied an online learning method called co-training to improve the local detection results. Cameras viewing the same scene exchange image patches are confident to be positive and negative examples of an object, comprising a few hundred bytes per frame. They implemented the algorithm on smart cameras and demonstrated that the approach resulted in increasingly better detection results.

The problem of matching objects between images from different cameras in a visual sensor network is sometimes called reacquisition or reidentification, and often results in a *handoff* of the responsibility of object tracking from one camera to another. Quaritsch et al. (Quaritsch, Kreuzthaler, Rinner, Bischof, and Strobl, 2007) described a distributed object tracking algorithm that illustrates the main ideas of decentralized tracking handoff. The basic idea is that the responsibility for tracking each object lies with a particular master camera. When the object nears the periphery of this camera's field of view, neighboring cameras are put into slave mode, searching for the object until it appears. When a slave discovers the object in its field of view, it becomes the new master.

Arth et al. (Arth, Leistner, and Bischof, 2007) proposed an algorithm for reacquisition based on signatures, or compact representations of local object features. The signatures are much less expensive to transmit through the network than the original images. The idea has some resemblance to the feature digest of Cheng et al. (Cheng, Devarajan, and Radke, 2007) discussed in Section 3. The algorithm was demonstrated on a simulated traffic camera network and its performance and communication properties with respect to noise analyzed in some detail. Park et al. (Park, Bhat, and Kak, 2006) proposed an algorithm for the distributed construction of look-up tables stored locally at each node that are used to select the cameras that are most likely to observe a specific location. These lookup tables could be used to improve camera handoff for multiple moving targets.

Iwaki et al. (Iwaki, Srivastava, Kosaka, Park, and Kak, 2008) pointed out the importance of developing tracking algorithms for large-scale camera networks that did not assume synchronized image capture. They presented an algorithm based on evidence accumulation that operates in a hierarchical manner. Cameras individually detect human heads in the scene and send the results to clusterheads for outlier detection and localization refinement.

Some researchers have suggested improvements in tracking by augmenting the input from visual sensors with non-visual sensor measurements. For example, Miyaki et al. (Miyaki, Yamasaki, and Aizawa, 2007) fused image tracking information with the signal strength from pedestrians' Wi-Fi mobile devices to better estimate their locations.

6 Conclusions

The past five years have seen an encouraging upturn in the development of distributed solutions to classical computer vision problems. We believe that there are many viable distributed alternatives for accurately estimating transition models and visual overlap, 3D calibration parameters, and 2D and 3D object tracks.

The highest-level algorithms surveyed here emphasize simple tracking rather than the estimation of more subtle object/environment characteristics. We view this as a natural next step in the development of distributed computer vision algorithms. As an early example, Chang and Aghajan (Chang and Aghajan, 2006) described fusion algorithms based on linear quadratic regression and Kalman filtering to estimate the orientation of faces imaged by a camera network.

Most algorithms also assume that the camera network is entirely engaged in performing the same vision task at every node, and that all tasks have the same priority. However, for distributed wireless camera networks deployed in the real world, it will be important to consider calibration and other computer vision tasks in the context of the many other tasks the node processors and network need to undertake. This will entail a tight coupling between the vision algorithms and the MAC, network, and link-layer protocols, organization, and channel conditions of the network, as well as the power supply, transmitter/receiver and kernel scheduler of each node. This tight integration would be critical for making the algorithms described here viable for embedded platforms, such as networks of cell-phone cameras (Bolliger, Köhler, and Römer, 2007). Bramberger et al. (Bramberger, Doblander, Maier, Rinner, and Schwabach, 2006) recently described an algorithm for allocating general tasks in a camera network, posing a distributed constraint satisfaction problem. When a smart camera poses a new task, the request is cycled through the cameras in the network, incrementally merging partial task allocations until the request arrives back at the originating camera.

Finally, we note that prototyping vision algorithms in a real network of cameras is a time-, labor-, and money-consuming proposition. Visually accurate, large-scale camera network simulation systems such as those developed by Qureshi and Terzopoulos (Qureshi and Terzopoulos, 2008) or Heath and Guibas (Heath and Guibas, 2008) will enable rapid prototyping of distributed computer vision algorithms without requiring the resources for an actual deployment.

References

Ai J, Abouzeid A (2006) Coverage by directional sensors in randomly deployed wireless sensor networks. Journal of Combinatorial Optimization 11(1):21 – 41

Arth C, Leistner C, Bischof H (2007) Object reacquisition and tracking in large-scale smart camera networks. In: First ACM/IEEE International Conference on Distributed Smart Cameras, pp 156–163

Avidan S, Moses Y, Moses Y (2007) Centralized and distributed multi-view correspondence. International Journal of Computer Vision 71(1):49–69

Baker P, Aloimonos Y (2000) Complete calibration of a multi-camera network. In: Proceedings of IEEE Workshop on Omnidirectional Vision 2000

Baker P, Aloimonos Y (2003) Calibration of a multicamera network. In: Proceedings of the IEEE Workshop on Omnidirectional Vision

Barton-Sweeney A, Lymberopoulos D, Savvides A (2006) Sensor localization and camera calibration in distributed camera sensor networks. In: Proceedings of the 3rd International Conference on Broadband Communications, Networks and Systems (BROADNETS), San Jose, CA, USA, pp 1–10

Bash BA, Desnoyers PJ (2007) Exact distributed Voronoi cell computation in sensor networks. In: International Conference on Information Processing in Sensor Networks

Bertsekas DP, Tsitsiklis JN (1997) Parallel and Distributed Computation: Numerical Methods. Athena Scientific

Bolliger P, Köhler M, Römer K (2007) Facet: Towards a smart camera network of mobile phones. In: Proceedings of Autonomics 2007 (ACM First International Conference on Autonomic Computing and Communication Systems), Rome, Italy

Boyd S, Ghosh A, Prabhakar B, Shah D (2005) Gossip algorithms: design, analysis and applications. In: IEEE INFOCOM, vol 3, pp 1653–1664

Bramberger M, Doblander A, Maier A, Rinner B, Schwabach H (2006) Distributed embedded smart cameras for surveillance applications. Computer 39(2):68–75

Brand M, Antone M, Teller S (2004) Spectral solution of large-scale extrinsic camera calibration as a graph embedding problem. In: Proceedings of the European Conference on Computer Vision

Capkun S, Hubaux JP (2005) Secure positioning of wireless devices with application to sensor networks. In: Proceedings of IEEE INFOCOM

Chang CC, Aghajan H (2006) Collaborative face orientation detection in wireless image sensor networks. In: Workshop on Distributed Smart Cameras

Chen X, Davis J, Slusallek P (2000) Wide area camera calibration using virtual calibration objects. In: IEEE Comp. Soc. Conf. on Computer Vision and Pattern Recognition

Cheng Z, Devarajan D, Radke R (2007) Determining vision graphs for distributed camera networks using feature digests. EURASIP Journal of Applied Signal Processing, Special Issue on Visual Sensor Networks Article ID 57034

Christopher C, Avi P (2003) Loopy belief propagation as a basis for communication in sensor networks. In: Proceedings of the 19th Annual Conference on Uncertainty in Artificial Intelligence (UAI-03), Morgan Kaufmann Publishers, San Francisco, CA, pp 159–166

Collins RT, Tsin Y (1999) Calibration of an outdoor active camera system. In: IEEE Computer Vision and Pattern Recognition 1999

Dellaert F, Kipp A, Krauthausen P (2005) A multifrontal QR factorization approach to distributed inference applied to multirobot localization and mapping. In: Proceedings of National Conference on Artificial Intelligence (AAAI 05), pp 1261–1266

Devarajan D, Radke R (2004) Distributed metric calibration for large-scale camera networks. In: Proceedings of the First Workshop on Broadband Advanced Sensor

Networks (BASENETS) 2004 (in conjunction with BroadNets 2004), San Jose, CA

Devarajan D, Radke R (2007) Calibrating distributed camera networks using belief propagation. EURASIP Journal of Applied Signal Processing, Special Issue on Visual Sensor Networks Article ID 60696

Devarajan D, Radke R, Chung H (2006) Distributed metric calibration of ad-hoc camera networks. ACM Transactions on Sensor Networks 2(3):380–403

Dockstader S, Tekalp A (2001) Multiple camera tracking of interacting and occluded human motion. Proceedings of the IEEE 89(10):1441–1455

Du W, Fang L, Peng N (2006) LAD: localization anomaly detection for wireless sensor networks. J Parallel Distrib Comput 66(7):874–886

Farrell R, Davis L (2008) Decentralized discovery of camera network topology. In: Second ACM/IEEE International Conference on Distributed Smart Cameras, Stanford University, Palo Alto, CA, USA

Farrell R, Doermann D, Davis L (2007) Learning higher-order transition models in medium-scale camera networks. In: IEEE 11th International Conference on Computer Vision

Funiak S, Guestrin C, Paskin M, Sukthankar R (2006) Distributed localization of networked cameras. In: Proceedings of the Fifth International Conference on Information Processing in Sensor Networks

Haas Z, et al (2002) Wireless ad hoc networks. In: Proakis J (ed) Encyclopedia of Telecommunications, John Wiley

Hartley R (1994) Self-calibration from multiple views with a rotating camera. In: Proc. ECCV '94, vol 1, pp 471–478

Hartley R, Zisserman A (2000) Multiple View Geometry in Computer Vision. Cambridge University Press

Heath K, Guibas L (2007) Facenet: Tracking people and acquiring canonical face images in a wireless camera sensor network. In: First ACM/IEEE International Conference on Distributed Smart Cameras, pp 117–124

Heath K, Guibas L (2008) Multi-person tracking from sparse 3D trajectories in a camera sensor network. In: Second ACM/IEEE International Conference on Distributed Smart Cameras, Stanford University, Palo Alto, CA, USA

van den Hengel A, Dick A, Detmold H, Cichowski A, Hill R (2007) Finding camera overlap in large surveillance networks. In: Proceedings of the 8th Asian Conference on Computer Vision

Hoffmann M, Hähner J (2007) Rocas: A robust online algorithm for spatial partitioning in distributed smart camera systems. In: First ACM/IEEE International Conference on Distributed Smart Cameras, pp 267–274

Ihler A, Fisher J, Moses R, Willsky A (2004) Nonparametric belief propagation for self-calibration in sensor networks. In: Proceedings of the 3rd International Symposium on Information Processing in Sensor Networks, Berkeley, CA

Iwaki H, Srivastava G, Kosaka A, Park J, Kak A (2008) A novel evidence accumulation framework for robust multi-camera person detection. In: Second ACM/IEEE International Conference on Distributed Smart Cameras, Stanford University, Palo Alto, CA, USA

Iyengar R, Sikdar B (2003) Scalable and distributed GPS free positioning for sensor networks. In: IEEE International Conference on Communications, vol 1, pp 338–342

Javed O, Rasheed Z, Shafique K, Shah M (2003) Tracking across multiple cameras with disjoint views. In: Proceedings of the 9th International Conference on Computer Vision, Nice, France

Javed O, Shafique K, Shah M (2005) Appearance modeling for tracking in multiple non-overlapping cameras. In: The IEEE Computer Society Conference on Computer Vision and Pattern Recognition

Jaynes C (1999) Multi-view calibration from planar motion for video surveillance. In: Second IEEE Workshop on Visual Surveillance (VS'99), pp 59–66

Kowalczyk W, Vlassis N (2004) Newscast EM. In: Neural Information Processing Systems Conference

Kulkarni P, Shenoy P, Ganesan D (2007) Approximate initialization of camera sensor networks. In: Proceedings of the 4th European Conference on Wireless Sensor Networks

Langendoen K, Reijers N (2003) Distributed localization in wireless sensor networks: A quantitative comparison. Computer Networks 43(4):499–518

Lee H, Aghajan H (2006) Collaborative node localization in surveillance networks using opportunistic target observations. In: Proceedings of the 4th ACM International Workshop on Video Surveillance and Sensor Networks, pp 9–18

Lee L, Romano R, Stein G (2000) Monitoring activities from multiple video streams: Establishing a common coordinate frame. IEEE Transactions on Pattern Recognition and Machine Intelligence 22(8)

Leistner C, Roth P, Grabner H, Bischof H, Starzacher A, Rinner B (2008) Visual online learning in distributed camera networks. In: Second ACM/IEEE International Conference on Distributed Smart Cameras, Stanford University, Palo Alto, CA, USA

Leonard J, Durrant-Whyte H (1991) Simultaneous map building and localization for an autonomous mobile robot. In: Proceedings of the IEEE/RSJ International Workshop on Intelligent Robots and Systems, pp 1442–1447

Liu X, Kulkarni P, Shenoy P, Ganesan D (2006) Snapshot: A self-calibration protocol for camera sensor networks. In: Proceedings of IEEE/CreateNet BASENETS 2006

Lowe DG (2004) Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision 60(2):91–110

Lynch N (1996) Distributed Algorithms. Morgan Kaufmann

Ma Y, Soatto S, Košecká J, Sastry S (2004) An Invitation to 3-D Vision: From Images to Geometric Models, Springer, chap 11

Maas HG (1999) Image sequence based automatic multi-camera system calibration techniques. ISPRS Journal of Photogrammetry and Remote Sensing 54(5-6):352–359

Makris D, Ellis T, Black J (2004) Bridging the gaps between cameras. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition

Mandel Z, Shimshoni I, Keren D (2007) Multi-camera topology recovery from coherent motion. In: First ACM/IEEE International Conference on Distributed Smart Cameras, pp 243–250

Mantzel W, Choi H, Baraniuk R (2004) Distributed camera network localization. In: Proceedings of the 38th Asilomar Conference on Signals, Systems and Computers

Marinakis D, Dudek G (2005) Topology inference for a vision-based sensor network. In: In Proc. of Canadian Conference on Computer and Robot Vision

Marinakis D, Dudek G (2006) Probabilistic self-localization for sensor networks. In: AAAI National Conference on Artificial Intelligence

Marinakis D, Dudek G (2008) Occam's razor applied to network topology inference. IEEE Transactions on Robotics 24(2):293 – 306

Medeiros H, Park J, Kak A (2007) A light-weight event-driven protocol for sensor clustering in wireless camera networks. In: First ACM/IEEE International Conference on Distributed Smart Cameras, pp 203–210

Medeiros H, Iwaki H, Park J (2008) Online distributed calibration of a large network of wireless cameras using dynamic clustering. In: Second ACM/IEEE International Conference on Distributed Smart Cameras, Stanford University, Palo Alto, CA, USA

Meingast M, Oh S, Sastry S (2007) Automatic camera network localization using object image tracks. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV) Workshop on Visual Representations and Modeling of Large-scale Environments

Meingast M, Kushwaha M, Oh S, Koutsoukos X, Ledeczi A, Sastry S (2008) Heterogeneous camera network localization using data fusion. In: Second ACM/IEEE International Conference on Distributed Smart Cameras, Stanford University, Palo Alto, CA, USA

Mensink T, Zajdel W, Krose B (2007) Distributed EM learning for appearance based multi-camera tracking. In: First ACM/IEEE International Conference on Distributed Smart Cameras, pp 178–185

Miyaki T, Yamasaki T, Aizawa K (2007) Multi-sensor fusion tracking using visual information and wi-fl location estimation. In: First ACM/IEEE International Conference on Distributed Smart Cameras, pp 275–282

Murphy KP, Weiss Y, Jordan MI (1999) Loopy belief propagation for approximate inference: An empirical study. In: Proceedings of Uncertainty in Artificial Intelligence (UAI '99), pp 467–475

Niu C, Grimson E (2006) Recovering non-overlapping network topology using far-field vehicle tracking data. In: Proceedings of the 18th International Conference on Pattern Recognition

Nowak R (2003) Distributed EM algorithms for density estimation and clustering in sensor networks. IEEE Transactions on Signal Processing 51(8):2245–2253

Park J, Bhat PC, Kak AC (2006) A look-up table based approach for solving the camera selection problem in large camera networks. In: Workshop on Distributed Smart Cameras

Paskin MA, Guestrin CE (2004) Robust probabilistic inference in distributed systems. In: Proceedings of the Twentieth Conference on Uncertainty in Artificial Intelligence (UAI '04), pp 436–445

Paskin MA, Guestrin CE, McFadden J (2005) A robust architecture for inference in sensor networks. In: 4th International Symposium on Information Processing in Sensor Networks (IPSN '05)

Qu W, Schonfeld D, Mohamed M (2007) Distributed Bayesian multiple-target tracking in crowded environments using multiple collaborative cameras. EURASIP Journal on Advances in Signal Processing 2007(38373)

Quaritsch M, Kreuzthaler M, Rinner B, Bischof H, Strobl B (2007) Autonomous multicamera tracking on embedded smart cameras. EURASIP Journal on Advances in Signal Processing 2007(92827)

Qureshi F, Terzopoulos D (2008) Smart camera networks in virtual reality. Proceedings of the IEEE

Rabbat M, Nowak R (2004) Distributed optimization in sensor networks. In: International Conference on Information Processing in Sensor Networks

Rahimi A, Dunagan B, Darrell T (2004) Simultaneous calibration and tracking with a network of non-overlapping sensors. In: The IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol 1, pp 187–194

Rao B, Durrant-Whyte H (1993) A decentralized Bayesian algorithm for identification of tracked targets. IEEE Transactions on Systems, Man and Cybernetics 23(6):1683–1698

Rekleitis I, Meger D, Dudek G (2006) Simultaneous planning, localization, and mapping in a camera sensor network. Robotics and Autonomous Systems 54(11):921–932

Rekleitis IM, Dudek G (2005) Automated calibration of a camera sensor network. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, pp 401–406

Rinner B, Winkler T, Schriebl W, Quaritsch M, Wolf W (2008) The evolution from single to pervasive smart cameras. In: Second ACM/IEEE International Conference on Distributed Smart Cameras, Stanford University, Palo Alto, CA, USA

Sastry N, Shankar U, Wagner D (2003) Secure verification of location claims. In: Proceedings of WiSe '03, the 2nd ACM workshop on Wireless Security, pp 1–10

Stein GP (1995) Accurate internal camera calibration using rotation, with analysis of sources of error. In: ICCV, pp 230–236

Taylor CJ, Shirmohammadi B (2006) Self localizing smart camera networks and their applications to 3D modeling. In: Proceedings of the International Workshop on Distributed Smart Cameras

Thorpe J, McEliece R (2002) Data Fusion Algorithms for Collaborative Robotic Exploration. Interplanetary Network Progress Report 149:1–14

Thrun S, Burgard W, Fox D (2005) Probabilistic Robotics: Intelligent Robotics and Autonomous Agents. MIT Press

Tieu K, Dalley G, Grimson W (2005) Inference of non-overlapping camera network topology by measuring statistical dependence. In: Tenth IEEE International Conference on Computer Vision, vol 2, pp 1842–1849

Triggs B, McLauchlan P, Hartley R, Fitzgibbon A (2000) Bundle adjustment – A modern synthesis. In: Triggs W, Zisserman A, Szeliski R (eds) Vision Algorithms: Theory and Practice, LNCS, Springer Verlag, pp 298–375

Tsai R (1992) A versatile camera calibration technique for high-accuracy 3-D machine vision metrology using off-the-shelf TV cameras and lenses. In: Wolff L, Shafer S, Healey G (eds) Radiometry – (Physics-Based Vision), Jones and Bartlett

Varshney PK (1997) Distributed Detection and Data Fusion. Springer

Venkatesh S, Alanyali M, Savas O (2006) Distributed detection in sensor networks with packet losses and finite capacity links. IEEE Transactions on Signal Processing 54(11):4118–4132

Wang B, Sung K, Ng T (2002) The localized consistency principle for image matching under non-uniform illumination variation and affine distortion. In: European Conference on Computer Vision, vol LNCS 2350, pp 205–219, copenhagen, Denmark

Wei Y, Yu Z, Guan Y (2007) Location verification algorithms for wireless sensor networks. In: Proceedings of the 27th International Conference on Distributed Computing Systems, p 70

Zou X, Bhanu B, Song B, Roy-Chowdhury A (2007) Determining topology in a distributed camera network. In: Proceedings of the IEEE International Conference on Image Processing